

Weekly Report

06/28/2015-07/04/2015


Research

AngularJS


I have worked through AngularJS's tutorial and build a simple web site.

Search:


Sort by: Newest ▼




Motorola XOOM™ with Wi-Fi
The Next, Next Generation Experience the future with Motorola XOOM with Wi-Fi, the world's first tablet powered by Android 3.0 (Honeycomb).



MOTOROLA XOOM™
The Next, Next Generation Experience the future with MOTOROLA XOOM, the world's first tablet powered by Android 3.0 (Honeycomb).



MOTOROLA ATRIX™ 4G
MOTOROLA ATRIX 4G the world's most powerful smartphone.



Dell Streak 7
Introducing Dell™ Streak 7. Share photos, videos and movies together. It's small enough to carry around, big enough to gather around.

TagCluster

This week, I have read the 7th chapter of 《社会计算：用户在线行为分析与挖掘》, which demonstrates social bookmarking analysis. This chapter mainly discuss how to cluster tags

according to users' tagging behavior. There are two problems in tag system: the first is resource are always tagged in different tags which is bad for resource sharing, the second problem is users need a recommendation when they want to chose a word to tagging their resource. In order to solve these problems, we need to cluster tags describing the same resource. Before clustering, we are supposed to define the similarity between two tags. We can define the similarity using Jaccard similarity coefficient.

$$J(t_i, t_j) = \frac{|O(t_i) \cap O(t_j)|}{|O(t_i) \cup O(t_j)|}$$

where $O(\alpha)$ means the set of resources connecting to α . However, Jaccard similarity coefficient only concen about direct connection(Fig. tag 1 and tag 2 connect to the same resouce). There is a Indirect connection between tag1 and tag3(existing a way from tag1 to tag3).

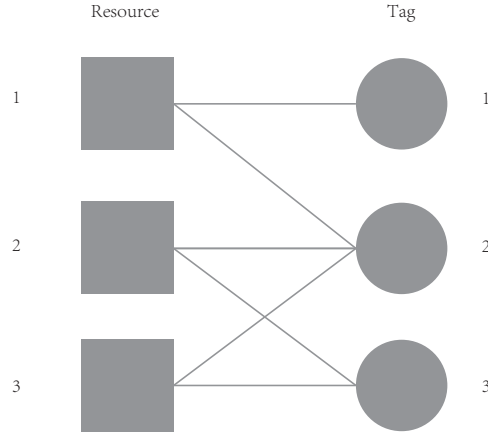


Figure 1: Indirect connection and direct connection

Consequently, we adopt a random walk to calculate similarity. Here, we calculate the probability of the way connecting a and b in $k+1$ steps. We walk a setp from a and b to one of their neighbours(x and y). t_{ax} is the probability from tag a to resource x.

$$\begin{aligned} P_{k+1}(a, b) &= \sum_{x \in O(a)} \sum_{y \in O(b)} t_{ax} t_{by} P_k(x, y) & a \neq b \\ P_{k+1}(a, b) &= 0 & a = b \end{aligned} \quad (1)$$

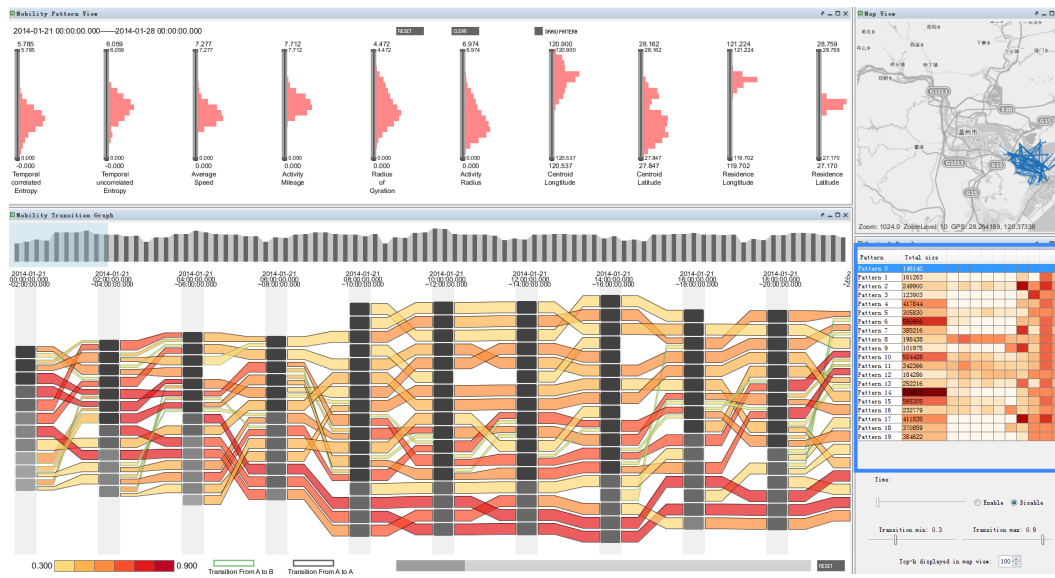
Now we know the similarity or distance between tags, we can cluster tag according to distance. We also define the similarity between tag t and cluster C .

$$R(t, C) = \begin{cases} \sum_{\alpha \in C} R(t, \alpha) / n & t \notin C \\ \sum_{\alpha \in C, t \neq \alpha} R(t, \alpha) / (n - 1) & t \in C \end{cases} \quad (2)$$

The TagClus algorithm is similar to DBSCAN. Given a constance β , α is a neighbour of t if $R(t, \alpha) \geq \beta$. We add the tag with biggest similarity into one cluster continually untill there is no neighbour to be added.

Mobility Pattern

I have changed number of clusters from 50 to 20 and I need more time to evaluate the result. Besides, in the 7th chapter of 《社会计算：用户在线行为分析与挖掘》, author use silhouette coefficient(SC) to evaluate the result of clustering. We can evaluate the result of k-means in different k to chose a best parameter.



Plan for next week

- Evaluate the result of k-means in different k using silhouette coefficient.